

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of: Joseph J. Pantuso et al.

Application No.: 10/071,587

Group No.: 2194

Filed: 02/08/2002

Examiner: Truong, Lechi

For: EXTRACTOR SYSTEM, METHOD AND COMPUTER PROGRAM PRODUCT FOR
MANAGING NETWORK ACCESS ON A PER-APPLICATION BASIS

Mail Stop Appeal Briefs – Patents

Commissioner for Patents

P.O. Box 1450

Alexandria, VA 22313-1450

**TRANSMITTAL OF APPEAL BRIEF
(PATENT APPLICATION--37 C.F.R. § 41.37)**

1. Transmitted herewith is an Appeal Brief in this application, with respect to the Notice of Appeal filed November 20, 2007, which reinstates the appeal originally instated by the Notice of Appeal filed on June 29, 2005, and the original Appeal Brief filed July 13, 2005.

2. STATUS OF APPLICANT

This application is on behalf of other than a small entity.

3. FEE FOR FILING APPEAL BRIEF

Pursuant to 37 C.F.R. § 41.20(b)(2), the fee for filing the Appeal Brief is:

other than a small entity	\$510.00
Appeal Brief fee due	\$510.00

4. EXTENSION OF TERM

The proceedings herein are for a patent application and the provisions of 37 C.F.R. § 1.136 apply.

Applicant believes that no extension of term is required. However, this conditional petition is being made to provide for the possibility that applicant has inadvertently overlooked the need for a petition and fee for extension of time.

5. TOTAL FEE DUE

The total fee due is:

Appeal brief fee	\$0.00 (previously paid on 07/13/2005)
Extension fee (if any)	\$0.00
TOTAL FEE DUE	\$0.00

6. FEE PAYMENT

Applicant believes that no fees are due in connection with the filing of this paper because the appeal brief fee was paid with a previous submission. However, the Commissioner is authorized to charge any additional fees that may be due (e.g. for any reason including, but not limited to, fee changes, etc.) to Deposit Account No. 50-1351 (Order No. NAIIP096).

7. FEE DEFICIENCY

If any additional extension and/or fee is required, and if any additional fee for claims is required, charge Deposit Account No. 50-1351 (Order No. NAIIP096).

Date: January 22, 2008

/KEVINZILKA/

Signature of Practitioner

Reg. No.: 41,429

Kevin J. Zilka

Tel. No.: 408-971-2573

Zilka-Kotab, PC

Customer No.: 28875

P.O. Box 721120

San Jose, CA 95172-1120

PATENT

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of:)	
)	
Pantuso et al.)	Group Art Unit: 2194
)	
Application No. 10/071,587)	Examiner: Truong, LeChi
)	
Filed: February 08, 2002)	Atty. Docket No.
)	NAIIP096/02.015.01
For: EXTRACTOR SYSTEM, METHOD)	
AND COMPUTER PROGRAM PRODUCT)	Date: 01/22/2008
FOR MANAGING NETWORK ACCESS)	
ON A PER-APPLICATION BASIS)	
<hr/>		

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

ATTENTION: Board of Patent Appeals and Interferences

APPEAL BRIEF (37 C.F.R. § 41.37)

Transmitted herewith is an Appeal Brief in this application, with respect to the Notice of Appeal filed November 20, 2007, which reinstates the appeal originally instated by the Notice of Appeal filed on June 29, 2005, and the original Appeal Brief filed July 13, 2005.

The fees required under § 1.17, and any required petition for extension of time for filing this brief and fees therefor, are dealt with in the accompanying TRANSMITTAL OF APPEAL BRIEF.

This brief contains these items under the following headings, and in the order set forth below (37 C.F.R. § 41.37(c)(i)):

- I REAL PARTY IN INTEREST
- II RELATED APPEALS AND INTERFERENCES
- III STATUS OF CLAIMS

- IV STATUS OF AMENDMENTS
- V SUMMARY OF CLAIMED SUBJECT MATTER
- VI GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL
- VII ARGUMENT
- VIII CLAIMS APPENDIX
- IX EVIDENCE APPENDIX
- X RELATED PROCEEDING APPENDIX

The final page of this brief bears the practitioner's signature.

I REAL PARTY IN INTEREST (37 C.F.R. § 41.37(c)(1)(i))

The real party in interest in this appeal is McAfee, Inc.

II RELATED APPEALS AND INTERFERENCES (37 C.F.R. § 41.37(c) (1)(ii))

With respect to other prior or pending appeals, interferences, or related judicial proceedings that will directly affect, or be directly affected by, or have a bearing on the Board's decision in the pending appeal, a prior appeal was noted on June 29, 2005, and was reinstated on December 12, 2005, and March 13, 2007, in the present application.

A Related Proceedings Appendix is appended hereto.

III STATUS OF CLAIMS (37 C.F.R. § 41.37(c) (1)(iii))

A. TOTAL NUMBER OF CLAIMS IN APPLICATION

Claims in the application are: 1-29

B. STATUS OF ALL THE CLAIMS IN APPLICATION

1. Claims withdrawn from consideration: None
2. Claims pending: 1-29
3. Claims allowed: None
4. Claims rejected: 1-29
5. Claims cancelled: None

C. CLAIMS ON APPEAL

The claims on appeal are: 1-29

See additional status information in the Appendix of Claims.

IV STATUS OF AMENDMENTS (37 C.F.R. § 41.37(c)(1)(iv))

As to the status of any amendment filed subsequent to final rejection, the Amendment submitted on March 30, 2005 was entered by the Examiner.

V SUMMARY OF CLAIMED SUBJECT MATTER (37 C.F.R. § 41.37(c)(1)(v))

With respect to a summary of Claim 1, as shown in Figure 5 et al., a method is provided for management of network access on a per application basis. In use, applications are selected from a group of applications adapted for working in conjunction with a first application program interface to gain access to a network (e.g. item 501 of Figure 5, etc.), the first application program interface adapted for permitting the applications to gain access to the network. Additionally, a second application program interface adapted for precluding the applications from accessing the network is installed (e.g. item 504 of Figure 5, etc.). Further, the selected applications are wrapped for allowing the selected applications to access the network via the second application program interface (e.g. item 506 of Figure 5, etc.), where the selected applications would otherwise be precluded network access by the second application program interface. See, for example, page 5, lines 3-11; and page 5, line 23 – page 6, line 2 et al.

With respect to a summary of Claim 8, as shown in Figure 5 et al., a computer program product is provided for management of network access on a per application basis. In use, computer code is provided for selecting applications from a group of applications adapted for working in conjunction with a first application program interface to gain access to a network (e.g. item 501 of Figure 5, etc.), the first application program interface adapted for permitting the applications to gain access to the network. Additionally, computer code is provided for installing a second application program interface adapted for precluding the applications from accessing the network (e.g. item 504 of Figure 5, etc.). Further, computer code is provided for wrapping the selected applications for allowing the selected applications to access the network via the second application program interface (e.g. item 506 of Figure 5, etc.), where the selected applications would otherwise be precluded network access by the second application program interface. See, for example, page 5, lines 3-11; and page 5, line 23 – page 6, line 2 et al.

With respect to a summary of Claim 15, as shown in Figure 5 et al., a system is provided for management of network access on a per application basis. In use, logic is provide for selecting applications from a group of applications adapted for working in conjunction with a first application program interface to gain access to a network (e.g. item 501 of Figure 5, etc.), the first application program interface adapted for permitting the applications to gain access to the

network . Additionally, logic is provided for installing a second application program interface adapted for precluding the applications from accessing the network (e.g. item 504 of Figure 5, etc.). Further, logic is provided for wrapping the selected applications for allowing the selected applications to access the network via the second application program interface (e.g. item 506 of Figure 5, etc.), where the selected applications would otherwise be precluded network access by the second application program interface. See, for example, page 5, lines 3-11; and page 5, line 23 – page 6, line 2 et al.

With respect to a summary of Claim 22, as shown in Figures 2 and 5 et al., a system is provided for management of network access on a per application basis. In use, means is provided for selecting applications (e.g. item 210 of Figure 2, etc.) from a group of applications adapted for working in conjunction with a first application program interface to gain access to a network (e.g. item 501 of Figure 5, etc.), the first application program interface adapted for permitting the applications to gain access to the network. Additionally, means is provided for installing (e.g. item 210 of Figure 2, etc.) a second application program interface adapted for precluding the applications from accessing the network (e.g. item 504 of Figure 5, etc.). Further, means is provided for wrapping (e.g. item 210 of Figure 2, etc.) the selected applications for allowing the selected applications to access the network via the second application program interface (e.g. item 506 of Figure 5, etc.), where the selected applications would otherwise be precluded network access by the second application program interface. See, for example, page 5, lines 3-11; page 5, line 23 – page 6, line 2; and page 8, lines 23-25 et al.

With respect to a summary of Claim 23, as shown in Figures 2 and 3 et al., a data structure stored in memory is provided for management of network access on a per application basis. In use, an application program interface object is provided (e.g. item 320' of Figure 3, etc.) for precluding a plurality of applications from accessing a network (e.g. item 235 of Figure 2, etc.). Also, a permitting application program interface (e.g. item 320 of Figure 3, etc.) is adapted for permitting the applications to gain access to the network. Additionally, a wrapper object is provided (e.g. item 322 of Figure 3, etc.) for wrapping selected applications (e.g. item 304 of Figure 3, etc.) for allowing the selected applications to access the network via the application program interface object, where the selected applications would otherwise be precluded network

access by the application program interface object. See, for example, page 5, lines 3-11; page 5, line 23 – page 6, line 2; and page 10, lines 7-20 et al.

With respect to a summary of Claim 24, as shown in Figure 5 et al., a method is provided for management of network access on a per application basis. In use, a precluding application program interface adapted for precluding a plurality of applications from accessing a network is installed (e.g. item 504 of Figure 5, etc.), wherein a permitting application program interface is adapted for permitting the applications to gain access to the network (e.g. item 501 of Figure 5, etc.). Additionally, a plurality of selected applications is wrapped for allowing the selected applications to access the network via the precluding application program interface (e.g. item 506 of Figure 5, etc.), where the selected applications would otherwise be precluded network access by the precluding application program interface. See, for example, page 5, lines 3-11; and page 5, line 23 – page 6, line 2 et al.

Of course, the above citations are merely examples of the above claim language and should not be construed as limiting in any manner.

VI GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL (37 C.F.R. § 41.37(c)(1)(vi))

Following, under each issue listed, is a concise statement setting forth the corresponding ground of rejection.

Issue # 1: The Examiner has rejected Claims 15-22 under 35 U.S.C. 101 as being directed toward non-statutory subject matter.

Issue # 2: The Examiner has rejected Claims 1, 8, 15, and 22-24 under 35 U.S.C. 103(a) as being unpatentable over Lee et al. (U.S. Publication No. 2003/0135465 A1), in view of Leahy et al. (U.S. Publication No. 2004/0133478 A1), and further in view of Pagé et al. (U.S. Patent No. 5,812,768).

Issue #3: The Examiner has rejected Claim 2 under 35 U.S.C. 103(a) as being unpatentable over Lee et al. (U.S. Publication No. 2003/0135465 A1), in view of Leahy et al. (U.S. Publication No. 2004/0133478 A1), in view of Pagé et al. (U.S. Patent No. 5,812,768), and further in view of OPT (Optimizations).

Issue #4: The Examiner has rejected Claims 3-6, 9-13, 16-20, and 25 under 35 U.S.C. 103(a) as being unpatentable over Lee et al. (U.S. Publication No. 2003/0135465 A1), in view of Leahy et al. (U.S. Publication No. 2004/0133478 A1), in view of Pagé et al. (U.S. Patent No. 5,812,768), in view of OPT (Optimizations), and further in view of Moeller (U.S. Patent No. 5,473,777).

Issue #5: The Examiner has rejected Claims 7, 14, and 21 under 35 U.S.C. 103(a) as being unpatentable over Lee et al. (U.S. Publication No. 2003/0135465 A1), in view of Leahy et al. (U.S. Publication No. 2004/0133478 A1), in view of Pagé et al. (U.S. Patent No. 5,812,768), and further in view of Alexander et al. (U.S. Patent No. 6,748,343 B2).

Issue #6: The Examiner has rejected Claims 26 and 29 under 35 U.S.C. 103(a) as being unpatentable over Lee et al. (U.S. Publication No. 2003/0135465 A1), in view of Leahy et al.

(U.S. Publication No. 2004/0133478 A1), in view of Pagé et al. (U.S. Patent No. 5,812,768), and further in view of Sitbon et al. (U.S. Patent No. 5,568,487).

Issue #7: The Examiner has rejected Claim 27 under 35 U.S.C. 103(a) as being unpatentable over Lee et al. (U.S. Publication No. 2003/0135465 A1), in view of Leahy et al. (U.S. Publication No. 2004/0133478 A1), in view of Pagé et al. (U.S. Patent No. 5,812,768), and further in view of Michael Norton (Basics of Network Segmentation: Switching and Bridging).

Issue #8: The Examiner has rejected Claim 28 under 35 U.S.C. 103(a) as being unpatentable over Lee et al. (U.S. Publication No. 2003/0135465 A1), in view of Leahy et al. (U.S. Publication No. 2004/0133478 A1), in view of Pagé et al. (U.S. Patent No. 5,812,768), and further in view of Bermudez et al. (U.S. Patent No. 6,874,149).

VII ARGUMENT (37 C.F.R. § 41.37(c)(1)(vii))

The claims of the groups noted below do not stand or fall together. In the present section, appellant explains why the claims of each group are believed to be separately patentable.

Issue # 1:

The Examiner has rejected Claims 15-22 under 35 U.S.C. 101 as being directed toward non-statutory subject matter.

Group #1: Claims 15-22

The Examiner has rejected Claims 15-22 under 35 U.S.C. 101 as being directed to non-statutory subject matter. More specifically, the Examiner has stated that “[s]electing, installing, [and] wrapping appear to be software functions” and that “[t]herefore, claims 15, 22 are non-statutory because they recite system claims that comprise software per se embodiments.”

Appellant respectfully disagrees and points out appellant’s claimed “logic” (Claim 15) and “means” (Claim 22) for completing appellant’s specific claim language.

Issue # 2:

The Examiner has rejected Claims 1, 8, 15, and 22-24 under 35 U.S.C. 103(a) as being unpatentable over Lee et al. (U.S. Publication No. 2003/0135465 A1), in view of Leahy et al. (U.S. Publication No. 2004/0133478 A1), and further in view of Pagé et al. (U.S. Patent No. 5,812,768).

Group # 1: Claims 1, 8, 15, and 22

To establish a *prima facie* case of obviousness, three basic criteria must be met. First, there must be some suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify the reference or to combine reference teachings. Second, there must be a reasonable expectation of success. Finally, the prior

art reference (or references when combined) must teach or suggest all the claim limitations. The teaching or suggestion to make the claimed combination and the reasonable expectation of success must both be found in the prior art and not based on appellant's disclosure. *In re Vaeck*, 947 F.2d 488, 20 USPQ2d 1438 (Fed.Cir.1991).

With respect to the first element of the *prima facie* case of obviousness and, in particular, the obviousness of combining the Lee and Leahy references, the Examiner argues that "[i]t would have been obvious to one of the ordinary skill in the art at the time the invention was made to modify the teaching of Lee to incorporate the feature[s] of network access [and] wrapping the selected applications for allowing the applications to access the network because this provide[s] predictability to the operator of an online commerce [site] as to user access [of] the online commerce site." To the contrary, appellant respectfully asserts that it would not have been obvious to combine the teachings of the Lee and Leahy references, especially in view of the vast evidence to the contrary.

More specifically, appellant respectfully notes that Lee discloses a "system and method... for securing content stored on media," and further teaches "attaching content privileges to the media" (Abstract – emphasis added). However, Leahy is teaches "[p]roviding prioritization of user online access to an online commerce site," and also discloses that "[t]hird party applications using API function calls to access an online commerce site are restricted to specific services by an access rule" (Abstract – emphasis added).

However, securing media content, as in Lee, and providing prioritized access to an online commerce site, as in Leahy, are clearly *non-analogous arts*. In view of the vastly different types of problems a content security system and an online commerce site prioritization system address, the Examiner's proposed combination is inappropriate.

One of the first inquiries in an obviousness analysis is whether all of the references relied on are analogous art. "In order to rely on a reference as a basis for rejection of an applicant's invention, the reference must either be in the field of applicant's endeavor or, if not, then be reasonably pertinent to the particular problem with which the inventor was concerned." *In re Oetiker*, 977 F.2d 1443, 1446, 24 USPQ2d 1443, 1445 (Fed. Cir. 1992).

Thus, appellant respectfully asserts that the first element of the *prima facie* case of obviousness has not been met, as noted above.

Moreover, appellant respectfully asserts that such references also fail to satisfy the third element of the *prima facie* case of obviousness. For example, with respect to the independent claims, the Examiner has relied on Paragraph [0141], lines 4-7; Paragraph [0142], lines 1-5; and Paragraph [0145], lines 5-8 from Lee, in addition to Paragraph [0014], lines 8-13 from Leahy, to make a prior art showing of appellant's claimed "selecting applications from a group of applications adapted for working in conjunction with a first application program interface to gain access to a network, the first application program interface adapted for permitting the applications to gain access to the network" (see this or similar, but not necessarily identical language in the independent claims).

Appellant respectfully asserts that the above excerpts from Lee merely disclose that "[a]ccess to secure content... is through open APIs and secure AP[I]s," that "[o]pen APIs...only allow plain file read and write access," and that "the secure metadata and unencrypted content are not available via open APIs" (Paragraph [0142], lines 1-5). In addition, the excerpts teach that "[t]he certificate... specifies the secure APIs... to which an application... may have access" (Paragraph [0145], lines 6-8). Further, the excerpts teach that "[t]he content elements... are encrypted" and that "when the content file... is read or copied through an open API an encrypted form is returned" (Paragraph [0141], lines 5-8). Further still, the above excerpt from Leahy merely teaches that "[t]he network... provides connectivity between any of the third party application servers... in the third party server area... and any of the API servers... of the online commerce site" (Paragraph [0014], lines 8-11).

However, merely disclosing that APIs are used to access secure content, that open APIs only allow plain file read and write access but cannot access secure metadata and unencrypted content, and that a certificate specifies the secure APIs that an application may access, as in Lee, in addition to disclosing that a network provides connectivity between application servers and API servers, as in Leahy, fails to even *suggest* "selecting applications from a group of applications," let alone where the "group of applications [is] adapted for working in conjunction

with a first application program interface to gain access to a network" (emphasis added), as claimed by appellant.

Appellant emphasizes that the excerpts from Lee relied on by the Examiner only relate to accessing content, and that the excerpt from Leahy relied on by the Examiner merely discloses connectivity between application servers and API servers. Clearly, such teachings do not even suggest appellant's claimed "selecting applications from a group of applications adapted for working in conjunction with a first application program interface to gain access to a network, the first application program interface adapted for permitting the applications to gain access to the network" (emphasis added), as claimed.

Furthermore, with respect to the independent claims, the Examiner has relied on Paragraph [0129], lines 1-2; Paragraph [0142], lines 7-8; and an unclear reference to the right Col., lines 14-19 from Lee, in addition to Paragraph [0014], lines 8-13 in Leahy, as excerpted below, to make a prior art showing of appellant's claimed "installing a second application program interface adapted for precluding the applications from accessing the network" (see this or similar, but not necessarily identical language in the independent claims).

"Engine 308 presents APIs to the players 310 and TPDRMs and CKDRMs to support a plurality of the usage..." (Lee-Paragraph [0129], lines 1-2 - emphasis added)

"The secure APIs 508 have restricted access to content 502 and 504 in that only secure APIs may retrieve..." (Lee-Paragraph [0142], lines 7-8 - emphasis added)

"Block level access to content on a hard drive of a computer is generally accessible via only a block driver. The firmware prevents open access by not supporting block access for block drivers. The file system may be hierarchical in nature. In one embodiment, the media disk may have a writeable, write-once portion, and a read-only..." (Lee-"right col", lines 14-19 - emphasis added)

"The network 130 provides connectivity between any of the third party application servers 110 and 115 in the third party server area 102 and any of the API servers 140, 142, and 144 of the online commerce site 106. Access by the third party application servers 110, 115 may be prioritized based on a service level agreement..." (Leahy-Paragraph [0014], lines 8-13)

Appellant respectfully asserts that the excerpts from Lee relied upon by the Examiner merely teach that an “[e]ngine 308 presents APIs to the players,” that “[t]he secure APIs 508 have restricted access to content,” and that “firmware prevents open access [to content] by not supporting block access for block drivers” (emphasis added). However, simply disclosing that APIs have restricted access to content, and that firmware prevents open access to content, as in Lee, fails to even suggest any sort of installation of an application program interface, as claimed, let alone “installing a second application program interface adapted for precluding the applications from accessing the network” (emphasis added), as claimed by appellant.

In addition, the excerpt from Leahy relied on by the Examiner only discloses “connectivity between any of the third party application servers...and any of the API servers.” Clearly, only disclosing connectivity between application servers and API servers fails to meet appellant’s claimed “installing a second application program interface adapted for precluding the applications from accessing the network” (emphasis added), as claimed.

Further, with respect to the independent claims, the Examiner has relied on Paragraph [0004], lines 4-7; Paragraph [0014], lines 8-13; Paragraph [0016], lines 2-10; Paragraph [0017], lines 15-20; Paragraph [0025], lines 1-13; and Paragraph [0042], lines 1-8 from Leahy to make a prior art showing of appellant’s claimed “wrapping the selected applications for allowing the selected applications to access the network via the second application program interface, where the selected applications would otherwise be precluded network access by the second application program interface” (see this or similar, but not necessarily identical language in the independent claims).

Appellant respectfully asserts that the excerpts relied upon by the Examiner merely teach that “an access rule may include a URL... that addresses the API server... with which the third party application... is to connect (or communicate) when accessing the services of the online commerce site” and that “[i]n this way, the URL directs a specific user to a specific API server on the online commerce site... per the service level agreement” (Paragraph [0016], lines 3-10 – emphasis added). Further, the excerpts teach “configur[ing] each access rule per the service level agreement to an API function used by a specific third party application to connect (via a given URL) to a specific API server” (Paragraph [0017], lines 15-17 – emphasis added).

Further still, the excerpts teach that “[a]n access rule defines which API server on the online commerce site a specific third party application may access when using a specific API function call” (Paragraph [0004], lines 4-7 – emphasis added). Additionally, the excerpts teach that “if the third party application request is validated, the access rule(s) for the predefined service levels for the identified third party application are returned to the requesting third party application” and that “[i]n this way, the third party application will be directed to the API server... providing the appropriate service level” (Paragraph [0025], lines 1-13 – emphasis added). Moreover, the excerpts disclose that “only those third party applications and API function calls that have the appropriate access rule may access the API server, thereby preventing other parties from sharing services by using a specific URL assigned to another third party application” and that “since a single third party application may access different API servers based on the specific API function call, the invention also prevents the third party user from arbitrary selecting any one of the API servers” (Paragraph [0042], lines 1-9 – emphasis added).

However, merely disclosing rules controlling to which API server a third party application connects when accessing the services of an online commerce site does not even *suggest* “wrapping the selected applications for allowing the selected applications to access the network,” much less “wrapping the selected applications for allowing the selected applications to access the network via the second application program interface, where the selected applications would otherwise be precluded network access by the second application program interface” (emphasis added), as claimed by appellant. Simply nowhere in the excerpts from Leahy relied on by the Examiner is there any disclosure of “a second application program interface adapted for precluding the applications from accessing the network,” where “the selected applications [are wrapped] for allowing the selected applications to access the network via the second application program interface,” in the context claimed.

Further, appellant respectfully points out that the Examiner admits that “Lee and Leahy do not explicitly teach wrapping the application for allowing the application [to] access the network via the application program interface.” However, the Examiner appears to also rely on Col. 3, lines 44-49 and Col. 47, line 63-Col. 48, line 3 of Page to make a prior art showing of appellant’s aforementioned claim language.

Appellant respectfully notes that the above excerpts relied on by the Examiner merely disclose that “[a]n adapter may also be provided as a gateway to convert a foreign communications protocol to the function server protocol to allow applications programs to access the service broker functionality even though they are not compatible with the application program interface” (Col.3, lines 44-48 – emphasis added). Additionally, the above excerpts teach that “[c]lient/server components that have not been written in accordance with the LAPI can be integrated through “adapters”” and that “[u]sing different adapters, the broker supports interoperability between programs that are otherwise unable to communicate with each other” (Col. 47, line 64-Col. 48, line 3).

However, merely disclosing the use of adapters to allow applications programs to access service broker functionality in no way teaches “wrapping the selected applications for allowing the selected applications to access the network via the second application program interface, where the selected applications would otherwise be precluded network access by the second application program interface” (emphasis added), as claimed.

Appellant respectfully asserts that at least the first and third elements of the *prima facie* case of obviousness have not been met, since it would not have been obvious to combine the prior art references, and since the prior art excerpts, as relied upon by the Examiner, fail to teach or suggest all of the claim limitations, as noted above.

Group #2: Claims 23 and 24

With respect to the independent claims, the Examiner has relied on Paragraph [0141], lines 5-7; Paragraph [0142], lines 7-8; Paragraph [0145], line 6-8; and an unclear reference to right Col., lines 14-19 in Lee, in addition to Paragraph [0014], lines 8-13 in Leahy to make a prior art showing of appellant’s claimed “application program interface object for precluding a plurality of applications from accessing a network, wherein a permitting application program interface is adapted for permitting the applications to gain access to the network” (see this or similar, but not necessarily identical language in the independent claims).

Appellant respectfully asserts that the above excerpts from Lee merely disclose that “[t]he certificate... specifies the secure APIs... to which an application... may have access” (Paragraph [0145], lines 6-8). Further, the excerpts teach that “[t]he content elements... are encrypted” and that “when the content file... is read or copied through an open API an encrypted form is returned” (Paragraph [0141], lines 5-8). In addition, the excerpts teach that “[t]he secure APIs 508 have restricted access to content” (Paragraph [0142], lines 7-8 - emphasis added),” and that “firmware prevents open access by not supporting block access for block drivers” (“right col”, lines 14-19 - emphasis added).

Further, the excerpt from Leahy discloses that “[t]he network... provides connectivity between any of the third party application servers... in the third party server area... and any of the API servers... of the online commerce site” (Paragraph [0014], lines 8-11).

Appellant respectfully asserts that simply disclosing secure APIs to which an application may have access, where such secure APIs have restricted access to content, as in Lee, in addition to connectivity between application servers API servers, as in Leahy, fails to meet appellant’s claimed “application program interface object for precluding a plurality of applications from accessing a network, wherein a permitting application program interface is adapted for permitting the applications to gain access to the network” (emphasis added), as claimed.

Also with respect to the independent claims, the Examiner has relied on Paragraph [0004], lines 4-7; Paragraph [0014], lines 8-13; Paragraph [0016], lines 2-10; Paragraph [0017], lines 15-20; Paragraph [0025], lines 1-13; and Paragraph [0042], lines 1-8 from Leahy to make a prior art showing of appellant’s claimed “a wrapper object for wrapping selected applications for allowing the selected applications to access the network via the application program interface object, where the selected applications would otherwise be precluded network access by the application program interface object” (see this or similar, but not necessarily identical language in the independent claims).

Appellant respectfully asserts that the excerpts relied upon by the Examiner merely teach that “an access rule may include a URL... that addresses the API server... with which the third party application... is to connect (or communicate) when accessing the services of the online

commerce site” and that “[i]n this way, the URL directs a specific user to a specific API server on the online commerce site... per the service level agreement” (Paragraph [0016], lines 3-10 – emphasis added). Further, the excerpts teach “configur[ing] each access rule per the service level agreement to an API function used by a specific third party application to connect (via a given URL) to a specific API server” (Paragraph [0017], lines 15-17 – emphasis added).

Further still, the excerpts teach that “[a]n access rule defines which API server on the online commerce site a specific third party application may access when using a specific API function call” (Paragraph [0004], lines 4-7 – emphasis added). Additionally, the excerpts teach that “if the third party application request is validated, the access rule(s) for the predefined service levels for the identified third party application are returned to the requesting third party application” and that “[i]n this way, the third party application will be directed to the API server... providing the appropriate service level” (Paragraph [0025], lines 1-13 – emphasis added). Moreover, the excerpts disclose that “only those third party applications and API function calls that have the appropriate access rule may access the API server, thereby preventing other parties from sharing services by using a specific URL assigned to another third party application” and that “since a single third party application may access different API servers based on the specific API function call, the invention also prevents the third party user from arbitrary selecting any one of the API servers” (Paragraph [0042], lines 1-9 – emphasis added).

However, merely disclosing rules controlling to which API server a third party application connects when accessing the services of an online commerce site does not even *suggest* “wrapping selected applications for allowing the selected applications to access the network,” much less “wrapping selected applications for allowing the selected applications to access the network via the application program interface object, where the selected applications would otherwise be precluded network access by the application program interface object” (emphasis added), as claimed by appellant. Simply nowhere in the excerpts from Leahy relied on by the Examiner is there any disclosure of “an application program interface object for precluding a plurality of applications from accessing a network,” where “selected applications [are wrapped] for allowing the selected applications to access the network via the application program interface object,” in the context claimed.

Further, appellant respectfully points out that the Examiner admits that “Lee and Leahy do not explicitly teach wrapping the application for allowing the application [to] access the network via the application program interface.” However, the Examiner appears to also rely on Col. 3, lines 44-49 and Col. 47, line 63-Col. 48, line 3 of Page to make a prior art showing of appellant’s aforementioned claim language.

Appellant respectfully notes that the above excerpts relied on by the Examiner merely disclose that “[a]n adapter may also be provided as a gateway to convert a foreign communications protocol to the function server protocol to allow applications programs to access the service broker functionality even though they are not compatible with the application program interface” (Col.3, lines 44-48 – emphasis added). Additionally, the above excerpts teach that “[c]lient/server components that have not been written in accordance with the LAPI can be integrated through “adapters” and that “[u]sing different adapters, the broker supports interoperability between programs that are otherwise unable to communicate with each other” (Col. 47, line 64-Col. 48, line 3).

However, merely disclosing the use of adapters to allow applications programs to access service broker functionality in no way teaches “wrapping selected applications for allowing the selected applications to access the network via the application program interface object, where the selected applications would otherwise be precluded network access by the application program interface object” (emphasis added), as claimed.

Appellant respectfully asserts that at least the first and third elements of the *prima facie* case of obviousness have not been met, since it would not have been obvious to combine the prior art references, and since the prior art excerpts, as relied upon by the Examiner, fail to teach or suggest all of the claim limitations, as noted above.

Issue #3:

The Examiner has rejected Claim 2 under 35 U.S.C. 103(a) as being unpatentable over Lee et al. (U.S. Publication No. 2003/0135465 A1), in view of Leahy et al. (U.S. Publication No.

2004/0133478 A1), in view of Pagé et al. (U.S. Patent No. 5,812,768), and further in view of OPT (Optimizations).

Group #1: Claim 2

With respect to the first element of the *prima facie* case of obviousness and, in particular, the obviousness of combining the Lee, Leahy, Pagé, and OPT references, the Examiner argues that “[i]t would have been obvious to one of the ordinary skill in the art at the time the invention was made to modify the teaching of Lee, Leahy and [Pagé] to incorporate the feature of a portable executable image because this decreases the image size and increases the program speed at a cost of increased link time.” To the contrary, appellant respectfully asserts that it would not have been obvious to combine the teachings of the Lee, Leahy, Pagé, and OPT references, especially in view of the vast evidence to the contrary.

More specifically, appellant respectfully points out that the OPT reference discloses “the optimizations that LINK performs during a build” (Page 1, Paragraph 1). Additionally, the OPT reference involves optimizing Windows 98© “[i]f you build images intended to run on Windows 98, especially those that are redistributable” (Page 2, Paragraph 6).

However, performing LINK optimizations for Windows 98©, as in OPT, is clearly *non-analogous art* with Lee, which relates to securing content (see Abstract), Leahy, which relates to online access to an online commerce site (see Abstract), and Pagé, which relates to inter-object communication (see Abstract). In view of the vastly different types of problems performing LINK optimizations addresses as opposed to securing content (Lee), online access (Leahy), and inter-object communication (Pagé), the Examiner’s proposed combination is inappropriate.

One of the first inquiries in an obviousness analysis is whether all of the references relied on are analogous art. “In order to rely on a reference as a basis for rejection of an applicant’s invention, the reference must either be in the field of applicant's endeavor or, if not, then be reasonably pertinent to the particular problem with which the inventor was concerned.” *In re Oetiker*, 977 F.2d 1443, 1446, 24 USPQ2d 1443, 1445 (Fed. Cir. 1992).

Thus, appellant respectfully asserts that the first element of the *prima facie* case of obviousness has not been met, as noted above.

Moreover, appellant respectfully asserts that such references also fail to satisfy the third element of the *prima facie* case of obviousness. For example, with respect to Claim 2, the Examiner has relied on Col. 3, lines 44-46 from Pagé, in addition to Page 2, lines 24-26 and Page 3, lines 16-19 from OPT, to make a prior art showing of appellant's claimed technique "wherein the selected applications are wrapped with a wrapper adapted for compressing data in a portable executable (PE) image that provides compression of data associated with the applications."

Appellant respectfully asserts that the excerpt from Pagé relied upon by the Examiner merely teaches that "[a]n adapter may also be provided as a gateway to convert a foreign communications protocol to the function server protocol to allow applications programs to access the service broker functionality even though they are not compatible with the application program interface" (Col.3, lines 44-48 – emphasis added). In addition, the excerpts from OPT relied on by the Examiner teach that "the value that is used to place the base of sections in the portable executable image... is 512 bytes by default" (Page 2, lines 23-25 – emphasis removed). In addition, the excerpts from OPT teach that "[t]he only time you should not use /OPT:WIN98 is when your portable executable image is very small" (Page 3, lines 16-17 – emphasis removed).

However, merely disclosing the conversion of a foreign communications protocol to a function server protocol, in addition to teaching that a portable executable image has a 512 byte default value and that a WIN98 optimization should not be used when a portable executable image is very small, in no way teaches a technique "wherein the selected applications are wrapped with a wrapper adapted for compressing data in a portable executable (PE) image that provides compression of data associated with the applications" (emphasis added), as claimed by appellant. Simply nowhere in the aforementioned reference excerpts are "the selected applications... wrapped with a wrapper adapted for compressing data in a portable executable (PE) image" (emphasis added), as claimed.

Again, appellant respectfully asserts that at least the first and third elements of the *prima facie* case of obviousness have not been met, since it would be *unobvious* to combine the references,

as noted above, and the prior art excerpts, as relied upon by the Examiner, fail to teach or suggest all of the claim limitations, as noted above.

Issue #4:

The Examiner has rejected Claims 3-6, 9-13, 16-20, and 25 under 35 U.S.C. 103(a) as being unpatentable over Lee et al. (U.S. Publication No. 2003/0135465 A1), in view of Leahy et al. (U.S. Publication No. 2004/0133478 A1), in view of Pagé et al. (U.S. Patent No. 5,812,768), in view of OPT (Optimizations), and further in view of Moeller (U.S. Patent No. 5,473,777).

Group #1: Claims 3-5 and 25

Appellant respectfully asserts that such claims are not met by the prior art for the reasons argued with respect to Issue #3, Group #1.

Group #2: Claims 6, 13 and 20

The Examiner has relied on the following Moeller excerpt to make a prior art showing of appellant's claimed "wherein the location in memory is where a routine is stored for allowing the selected applications to access the network" (see Claim 6 et al.).

"The library server processes the request by accessing the desired computer program logic from the code library and sending the desired computer program logic to the area of memory designated by the destination address." (Col. 9, lines 17-20)

Appellant respectfully asserts that such excerpt fails to even suggest any sort of "location in memory...where a routine is stored for allowing the selected applications to access the network" (emphasis added), as claimed. In particular, merely disclosing that a library server sends logic to an area of memory, as in Moeller, simply does not meet the specificity of appellant's claims, as noted above.

Appellant respectfully asserts that at least the third element of the *prima facie* case of obviousness has not been met, as noted above.

Group #3: Claim 9 and 16

With respect to the first element of the *prima facie* case of obviousness and, in particular, the obviousness of combining the Lee, Leahy, Pagé, Moeller and OPT references, the Examiner argues that “[i]t would have been obvious to one of the ordinary skill in the art at the time the invention was made to modify the teaching of Lee, Leahy and [Pagé] and OPT to incorporate the feature of extract[ing] the data in the PE image because this enables the object oriented application to access in an object oriented manner by using a native procedural interface.” To the contrary, appellant respectfully asserts that it would not have been obvious to combine the teachings of the Lee, Leahy, Pagé, Moeller and OPT references, especially in view of the vast evidence to the contrary.

More specifically, appellant respectfully points out that the OPT reference discloses “the optimizations that LINK performs during a build” (Page 1, Paragraph 1). Additionally, the OPT reference involves optimizing Windows 98© “[i]f you build images intended to run on Windows 98, especially those that are redistributable” (Page 2, Paragraph 6).

However, performing LINK optimizations for Windows 98©, as in OPT, is clearly *non-analogous art* with Lee, which relates to securing content (see Abstract), Leahy, which relates to online access to an online commerce site (see Abstract), Pagé, which relates to inter-object communication (see Abstract), and Moeller, which relates to enabling an object-oriented application to access in an object-oriented manner a procedural operating system (see Abstract). In view of the vastly different types of problems performing LINK optimizations addresses as opposed to securing content (Lee), online access (Leahy), inter-object communication (Pagé), and a procedural operating system access by an object-oriented application (Moeller), the Examiner’s proposed combination is inappropriate.

One of the first inquiries in an obviousness analysis is whether all of the references relied on are analogous art. “In order to rely on a reference as a basis for rejection of an applicant’s invention, the reference must either be in the field of applicant's endeavor or, if not, then be reasonably

pertinent to the particular problem with which the inventor was concerned.” *In re Oetiker*, 977 F.2d 1443, 1446, 24 USPQ2d 1443, 1445 (Fed. Cir. 1992).

Thus, appellant respectfully asserts that the first element of the *prima facie* case of obviousness has not been met, as noted above.

Moreover, appellant respectfully asserts that such references also fail to satisfy the third element of the *prima facie* case of obviousness. For example, with respect to Claim 2, the Examiner has relied on Col. 3, lines 44-46 from Pagé, in addition to Page 2, lines 24-26 and Page 3, lines 16-19 from OPT, to make a prior art showing of appellant’s claimed technique “wherein the selected applications are wrapped with a wrapper adapted for compressing data in a portable executable (PE) image that provides compression of data associated with the applications.”

Appellant respectfully asserts that the excerpt from Pagé relied upon by the Examiner merely teaches that “[a]n adapter may also be provided as a gateway to convert a foreign communications protocol to the function server protocol to allow applications programs to access the service broker functionality even though they are not compatible with the application program interface” (Col.3, lines 44-48 – emphasis added). In addition, the excerpts from OPT relied on by the Examiner teach that “the value that is used to place the base of sections in the portable executable image... is 512 bytes by default” (Page 2, lines 23-25 – emphasis removed). In addition, the excerpts from OPT teach that “[t]he only time you should not use /OPT:WIN98 is when your portable executable image is very small” (Page 3, lines 16-17 – emphasis removed).

However, merely disclosing the conversion of a foreign communications protocol to a function server protocol, in addition to teaching that a portable executable image has a 512 byte default value and that a WIN98 optimization should not be used when a portable executable image is very small, in no way teaches a technique “wherein the selected applications are wrapped with a wrapper adapted for compressing data in a portable executable (PE) image that provides compression of data associated with the applications” (emphasis added), as claimed by appellant. Simply nowhere in the aforementioned reference excerpts are “the selected applications... wrapped with a wrapper adapted for compressing data in a portable executable (PE) image” (emphasis added), as claimed.

Again, appellant respectfully asserts that at least the first and third elements of the *prima facie* case of obviousness have not been met, since it would be *unobvious* to combine the references, as noted above, and the prior art excerpts, as relied upon by the Examiner, fail to teach or suggest all of the claim limitations, as noted above.

Group # 4: Claims 10-12 and 17-19

Appellant respectfully asserts that such claims are not met by the prior art for the reasons argued with respect to Issue #4, Group #3.

Issue #5:

The Examiner has rejected Claims 7, 14, and 21 under 35 U.S.C. 103(a) as being unpatentable over Lee et al. (U.S. Publication No. 2003/0135465 A1), in view of Leahy et al. (U.S. Publication No. 2004/0133478 A1), in view of Pagé et al. (U.S. Patent No. 5,812,768), and further in view of Alexander et al. (U.S. Patent No. 6,748,343 B2).

Group #1: Claims 7, 14 and 21

The Examiner has relied on the following excerpt from Alexander to make a prior art showing of appellant's claimed "allowing a user to select the applications to be allowed to access the network via the second application program interface."

"...a computer display operable to generate a user interface for obtaining a user selection of client, premises, location, monitoring device, and processing rule data and to transmit the data to the processing server..." (Col. 19, lines 53-56)

Appellant respectfully asserts that the above excerpt from Alexander does not teach any sort of applications, as appellant claims, but instead only discloses selecting items associated with transmitting data, such as a client, location, etc. In addition, there is no mention in the above excerpt of any type of second application program interface, as claimed, especially where such "second application program interface [is] adapted for precluding the applications from

accessing the network,” in the context claimed (see independent claim for context). Simply nowhere in the above excerpt relied on by the Examiner is there any mention of “allowing a user to select the applications to be allowed to access the network via the second application program interface” (emphasis added), as claimed.

Appellant respectfully asserts that at least the third element of the *prima facie* case of obviousness has not been met, as noted above.

Issue #6:

The Examiner has rejected Claims 26 and 29 under 35 U.S.C. 103(a) as being unpatentable over Lee et al. (U.S. Publication No. 2003/0135465 A1), in view of Leahy et al. (U.S. Publication No. 2004/0133478 A1), in view of Pagé et al. (U.S. Patent No. 5,812,768), and further in view of Sitbon et al. (U.S. Patent No. 5,568,487).

Group #1: Claim 26

Appellant respectfully asserts that such claims are not met by the prior art for the reasons argued with respect to Issue #2, Group #1.

Group #2: Claim 29

The Examiner has relied on Col. 2, lines 25-30 in Sitbon to make a prior art showing of appellant’s claimed technique “wherein the second application program interface is separate from the first application program interface.”

Appellant respectfully asserts that the excerpt from Sitbon relied on by the Examiner simply teaches “passage from the TCP/IP protocol to the OSI/CO protocol” where “the ‘socket’ interface calls, and the system calls...intended for the TCP/IP network are converted and processed.” Clearly, simply disclosing converting calls between protocols, as in Sitbon, fails to even mention a first application program interface and a second application program interface, as

appellant claims, and especially not where “the second application program interface is separate from the first application program interface,” as claimed.

Appellant respectfully asserts that at least the first and third elements of the *prima facie* case of obviousness have not been met, since it would not have been obvious to combine the prior art references, and since the prior art excerpts, as relied upon by the Examiner, fail to teach or suggest all of the claim limitations, as noted above.

Issue #7:

The Examiner has rejected Claim 27 under 35 U.S.C. 103(a) as being unpatentable over Lee et al. (U.S. Publication No. 2003/0135465 A1), in view of Leahy et al. (U.S. Publication No. 2004/0133478 A1), in view of Pagé et al. (U.S. Patent No. 5,812,768), and further in view of Michael Norton (Basics of Network Segmentation: Switching and Bridging).

Group #1: Claim 27

The Examiner has relied on Michael Norton’s disclosure of a network card that attempts to transmit frames onto a wire (Consuming bandwidth: lines 4-5) to make a prior art showing of appellant’s claimed technique “wherein the second application program interface is adapted for precluding the applications from accessing the network utilizing a network card.”

Appellant respectfully asserts that Michael Norton simply discloses a network card that attempts to transmit frames, which does not meet, and even *teaches away* from, appellant’s specific claim language. In particular, the network card claimed by appellant is utilized for “the second application program interface...precluding the applications from accessing the network” (emphasis added), as claimed.

Appellant respectfully asserts that at least the first and third element of the *prima facie* case of obviousness has not been met, as noted above.

Issue #8:

The Examiner has rejected Claim 28 under 35 U.S.C. 103(a) as being unpatentable over Lee et al. (U.S. Publication No. 2003/0135465 A1), in view of Leahy et al. (U.S. Publication No. 2004/0133478 A1), in view of Pagé et al. (U.S. Patent No. 5,812,768), and further in view of Bermudez et al. (U.S. Patent No. 6,874,149).

Group #1: Claim 28

Appellant respectfully asserts that such claims are not met by the prior art for the reasons argued with respect to Issue #2, Group #1.

In view of the remarks set forth hereinabove, all of the independent claims are deemed allowable, along with any claims depending therefrom.

VIII CLAIMS APPENDIX (37 C.F.R. § 41.37(c)(1)(viii))

The text of the claims involved in the appeal (along with associated status information) is set forth below:

1. (Previously Presented) A method for management of network access on a per application basis, comprising:
 - (a) selecting applications from a group of applications adapted for working in conjunction with a first application program interface to gain access to a network, the first application program interface adapted for permitting the applications to gain access to the network;
 - (b) installing a second application program interface adapted for precluding the applications from accessing the network; and
 - (c) wrapping the selected applications for allowing the selected applications to access the network via the second application program interface, where the selected applications would otherwise be precluded network access by the second application program interface.
2. (Previously Presented) The method as recited in claim 1, wherein the selected applications are wrapped with a wrapper adapted for compressing data in a portable executable (PE) image that provides compression of data associated with the applications.
3. (Original) The method as recited in claim 2, wherein the wrapper equips the compressed data with extractor code adapted for extracting the data in the PE image.
4. (Original) The method as recited in claim 3, wherein the extractor code is further adapted for interfacing with the second application program interface.
5. (Original) The method as recited in claim 2, wherein the wrapper is further adapted for identifying a location in memory.
6. (Original) The method as recited in claim 5, wherein the location in memory is where a routine is stored for allowing the selected applications to access the network.

7. (Original) The method as recited in claim 1, and further comprising allowing a user to select the applications to be allowed to access the network via the second application program interface.
8. (Previously Presented) A computer program product for management of network access on a per application basis, comprising:
 - (a) computer code for selecting applications from a group of applications adapted for working in conjunction with a first application program interface to gain access to a network, the first application program interface adapted for permitting the applications to gain access to the network;
 - (b) computer code for installing a second application program interface adapted for precluding the applications from accessing the network; and
 - (c) computer code for wrapping the selected applications for allowing the selected applications to access the network via the second application program interface, where the selected applications would otherwise be precluded network access by the second application program interface.
9. (Previously Presented) The computer program product as recited in claim 8, wherein the selected applications are wrapped with a wrapper adapted for compressing data in a portable executable (PE) image that provides compression of data associated with the applications.
10. (Original) The computer program product as recited in claim 9, wherein the wrapper equips the compressed data with extractor code adapted for extracting the data in the PE image.
11. (Original) The computer program product as recited in claim 10, wherein the extractor code is further adapted for interfacing with the second application program interface.
12. (Original) The computer program product as recited in claim 9, wherein the wrapper is further adapted for identifying a location in memory.

13. (Original) The computer program product as recited in claim 12, wherein the location in memory is where a routine is stored for allowing the selected applications to access the network.
14. (Original) The computer program product as recited in claim 8, and further comprising computer code for allowing a user to select the applications to be allowed to access the network via the second application program interface.
15. (Previously Presented) A system for management of network access on a per application basis, comprising:
 - (a) logic for selecting applications from a group of applications adapted for working in conjunction with a first application program interface to gain access to a network, the first application program interface adapted for permitting the applications to gain access to the network;
 - (b) logic for installing a second application program interface adapted for precluding the applications from accessing the network; and
 - (c) logic for wrapping the selected applications for allowing the selected applications to access the network via the second application program interface, where the selected applications would otherwise be precluded network access by the second application program interface.
16. (Previously Presented) The system as recited in claim 15, wherein the selected applications are wrapped with a wrapper adapted for compressing data in a portable executable (PE) image that provides compression of data associated with the applications.
17. (Original) The system as recited in claim 16, wherein the wrapper equips the compressed data with extractor code adapted for extracting the data in the PE image.
18. (Original) The system as recited in claim 17, wherein the extractor code is further adapted for interfacing with the second application program interface.

19. (Original) The system as recited in claim 16, wherein the wrapper is further adapted for identifying a location in memory.
20. (Original) The system as recited in claim 19, wherein the location in memory is where a routine is stored for allowing the selected applications to access the network.
21. (Original) The system as recited in claim 15, and further comprising logic for allowing a user to select the applications to be allowed to access the network via the second application program interface.
22. (Previously Presented) A system for management of network access on a per application basis, comprising:
 - (a) means for selecting applications from a group of applications adapted for working in conjunction with a first application program interface to gain access to a network, the first application program interface adapted permitting the applications to gain access to the network;
 - (b) means for installing a second application program interface adapted for precluding the applications from accessing the network; and
 - (c) means for wrapping the selected applications for allowing the selected applications to access the network via the second application program interface, where the selected applications would otherwise be precluded network access by the second application program interface.
23. (Previously Presented) A data structure stored in memory for management of network access on a per application basis, comprising:
 - (a) application program interface object for precluding a plurality of applications from accessing a network, wherein a permitting application program interface is adapted for permitting the applications to gain access to the network; and
 - (b) a wrapper object for wrapping selected applications for allowing the selected applications to access the network via the application program interface object, where the selected applications would otherwise be precluded network access by the application program interface object.

24. (Previously Presented) A method for management of network access on a per application basis, comprising:
 - (a) installing a precluding application program interface adapted for precluding a plurality of applications from accessing a network, wherein a permitting application program interface is adapted for permitting the applications to gain access to the network; and
 - (b) wrapping a plurality of selected applications for allowing the selected applications to access the network via the precluding application program interface, where the selected applications would otherwise be precluded network access by the precluding application program interface.
25. (Previously Presented) The method as recited in claim 2, wherein the PE image includes a header, a stub program, a file signature, a .text section header, a .bss section header, a .rdata section header, and a .debug section header.
26. (Previously Presented) The method as recited in claim 1, wherein the applications include a word processor application, a database program, a browser program, a development tool program, a drawing program, an image editing program, and a communication program.
27. (Previously Presented) The method as recited in claim 1, wherein the second application program interface is adapted for precluding the applications from accessing the network utilizing a network card.
28. (Previously Presented) The method as recited in claim 1, wherein the second application program interface includes a modified copy of the first application program interface.
29. (Previously Presented) The method as recited in claim 1, wherein the second application program interface is separate from the first application program interface.

IX EVIDENCE APPENDIX (37 C.F.R. § 41.37(c)(1)(ix))

There is no such evidence.

X RELATED PROCEEDING APPENDIX (37 C.F.R. § 41.37(c)(1)(x))

Since no decision(s) has been rendered in such proceeding(s), no material is included in this Related Proceedings Appendix.

In the event a telephone conversation would expedite the prosecution of this application, the Examiner may reach the undersigned at (408) 971-2573. For payment of any additional fees due in connection with the filing of this paper, the Commissioner is authorized to charge such fees to Deposit Account No. 50-1351 (Order No. Order No. NA11P096).

Respectfully submitted,

By: /KEVINZILKA/ Date: January 22, 2008

Kevin J. Zilka
Reg. No. 41,429

Zilka-Kotab, P.C.
P.O. Box 721120
San Jose, California 95172-1120
Telephone: (408) 971-2573
Facsimile: (408) 971-4660